07.03.2023

# Homework 1

By Stamate Valentin

The implementation is made using the C/C++ programming language for the low level functions provided that makes it easier to read and write a buffer to a file descriptor. The server and client uses both TCP and UDP protocols to send and receive the packages. After the client connects to the server via IP address and the port, it communicates parameters such as the number of bytes of every package, the dataset that wants to get files from and a flag for confirmation of packages. The server finds all the files and for each one, packages with the size requested by the client are sent. The files are then reconstructed in the client in order to make a copy of them. Before sending each package first is send the number of bytes read from the files. After the files are send, the statistics are shown in the console and the connection is closed.

The TCP server can serve multiple clients at the same time. For each connection a new thread is opened. The UDP server only waits for connections. Both of them run in two docker containers in a virtual network and a volume is attatched to them that point to a host location where the files are located in order to save space.

Below are the statistic regarding the transfer. Only the client side statistics are used because the client would have them the same but reversed. The dataset used contains photos summing up 1GB.

TCP Acknowledge | Client <- Server

| Package Size | Time | Package Received | Packages Sent | Bytes Read | Bytes Sent |
|---|---|---|---|---|---|
| 1kB | 44000s | 2003572 | 101864 | 1037162689 | 4007200 |
| 4kB | 26s | 489652 | 244490 | 1003837391 | 977960 |
| 16kB/300us | 43s | 123015 | 61171 | 1003742353 | 244685 |
| 32kB/300us | 26s | 62054 | 30643 | 1000563537 | 122572 |
| 55KB | 14s | 37231 | 18350 | 1005436827 | 73400 |

TCP Streaming | Client <- Server

| Package Size | Time | Package Received | Packages Sent | Bytes Read | Bytes Sent |
|---|---|---|---|---|---|
| 1kB | 2315s | 2004354 | 3 | 1008969019 | 12 |
| 4kB | 543s | 489038 | 3 | 1004419757 | 12 |
| 16kB/1ms | 139s | 122936 | 3 | 1001582559 | 12 |
| 32kB/1ms | 68s | 62054 | 3 | 1000322495 | 12 |
| 55kB | 43s | 37231 | 3 | 1005436827 | 12 |

07.03.2023

## UDP Acknowledge | Client <- Server

| Package Size | Time | Package Received | Packages Sent | Bytes Read | Bytes Sent |
|---|---|---|---|---|---|
| 1kB | 40s | 1968837 | 984165 | 1011712832 | 3936660 |
| 4kB | 12s | 492697 | 246095 | 1008760552 | 984380 |
| 16kB/1ms | 5s | 123649 | 61571 | 1008022456 | 246284 |
| 32kB/1ms | 3s | 62141 | 30817 | 1007899440 | 62141 |
| 55kB | 25s | 37231 | 18350 | 1745508961 | 73400 |

## UDP Streaming | Client <- Server

| Package Size | Time | Package Received | Packages Sent | Bytes Read | Bytes Sent |
|---|---|---|---|---|---|
| 1kB | 2316s | 1951044 | 3 | 1002718648 | 12 |
| 4kB | 611s | 492697 | 3 | 1008760552 | 12 |
| 16kB/1ms | 150s | 123649 | 3 | 1008022456 | 12 |
| 32kB/1ms | 77s | 62141 | 3 | 1121368722 | 12 |
| 55kB | 46s | 37231 | 3 | 1005436827 | 12 |

This implementation is not working on local network.